CN 53-1189/P ISSN 1672-7673

瞬变源快速自动响应观测系统的研究与实现*

符夏川[1,2,3]，吴 潮[1,2]，黄茂海[1,2,3]，张 墨[1,2]，卢晓猛[1,2]

（1. 中国科学院国家天文台，北京 100012；
2. 中国科学院空间天文与技术重点实验室，北京 100012；3. 中国科学院大学，北京 100049）

摘要：天文瞬变源后随观测对于瞬变源证认具有重要的意义。瞬变源发生的偶然性和光变的快速性要求后随观测系统具有快速自动响应的能力。介绍了基于 Python 开发的天文瞬变源快速自动响应观测系统的研究和实现，核心是基于 Django 架构实现数据库交互与模块调度功能。系统有基于 VOEvent 的警报消息传递、网络端状态监控与交互平台、数据自动处理和结果自动反馈以及望远镜控制接口等主要功能。通过在北京的中法天文卫星科学中心和兴隆观测基地之间的消息数据传递和实际触发观测测试表明，系统能满足中法天文卫星项目地基观测系统瞬变源后随自动响应观测的基本要求，同时系统还具有良好的可移植性，适用于其他类似的自动响应观测系统。

关键词：天文数据处理；瞬变源；自动观测

中图分类号：TP311 **文献标识码**：A **文章编号**：1672-7673(2017)04-0452-09

天文瞬变源的观测与研究在现代时域天文中占据越来越重要的地位，如伽玛暴(Gamma Ray Burst，GRB)和引力波暴等瞬变源光学对应体的后随观测与搜索对研究瞬变源的物理属性有重要作用。因瞬变源发生具有随机性和突发性，且持续时间长短不一，这就要求后随观测设备具有快速实时响应瞬变源触发信息的能力，能及时将望远镜指向目标天区进行后随观测，并能够快速处理观测数据，将处理结果实时上传与发送到整个天文界，以利于更大口径望远镜设备的进一步后随观测。

中法天文卫星项目(Space-based multi-band astronomical Variable Objects Monitor，SVOM)[1]，其主要目标是伽玛暴的观测研究，以卫星上的多波段探测器和地面上的广角相机阵以及专用后随望远镜组成天文联合多波段观测网络。其中专用地基后随望远镜（Ground Follow-up Telescope，GFT）由中法双方各一台组成，分布在东西半球实现全天无间断观测覆盖。地基后随望远镜需要在收到警报信息后的 5 min 内进行观测，将观测结果自动上传远程的科学中心，并通报给整个天文界，便于国际上其他大口径望远镜及时决定是否需要进行快速的后随跟进观测。国际类似的自动望远镜有 BOOTES[2]，MASTER[3]，ROTSE[4]，LT[5]，RAPTOR[6]，但由于 SVOM 项目对数据处理过程和处理时间要求更高，在功能上要求更为全面，并且许多国际项目的运行软件并没有公开。本文结合科学和工程条件的实际情况，根据需求研究和实现了瞬变源快速自动响应观测系统。

瞬变源快速自动响应观测系统主要由以下功能模块组成：中央调度控制，警报和数据实时传输与监控，数据自动处理与管理，硬件控制接口以及用户监控与交互。利用 Python 的面向对象易于开发、具有丰富的天文软件包的优点，系统开发主要基于 Python 语言。系统在 Django 框架下开发与之配套的数据库，实现模块调度的参数传递、原数据及数据产品管理以及 Django 网页服务的用户交互支撑。系统主要具备警报触发、望远镜观测控制、数据处理及传输三大功能，以及基于网页服务的系统运行监控和交互功能。

* 基金项目：国家自然科学基金（11603049，U1431108）资助.
收稿日期：2017-03-09；修订日期：2017-04-11
作者简介：符夏川，男，硕士. 研究方向：天文技术与方法. Email：xcfu@nao.cas.cn
通讯作者：吴 潮，男，副研究员. 研究方向：天文大数据实时处理. Email：cwu@bao.ac.cn

论文主要就以下几方面进行阐述：(1)系统总体框架设计，主要描述系统的基本需求和设计思路以及关键技术与功能；(2)实时消息与数据传输，描述警报接收与数据回传网络设计和数据传输协议；(3)在 Django 框架下，基于数据库实现模块调度的参数传递、数据管理以及模块之间的关系与调度；(4)系统运行监控与交互；(5)系统在实际观测环境下的测试；(6)讨论并展望下一步工作。

1　系统总体框架设计

根据数据处理的流程分析，瞬变源快速自动响应观测系统需要完成的数据处理流程和模块组成架构如图 1，即从科学中心(或者国际警报网络)接收瞬变源警报，接收警报后警报监控模块对警报进行解释并判断是否可观测，然后生成观测计划，并通过控制接口对望远镜进行观测控制，同时，如果有其他常规观测，则中断常规观测。CCD 相机完成数据采集后，后台调控系统触发数据自动处理系统对数据进行处理，处理的结果自动反馈给远程科学中心。系统运行监控与交互主要通过网页服务实现用户对系统运行状态的人工检验以及观测计划的人工干预等。本文主要研究开发了警报监控、望远镜控制接口、原始数据接口、数据 FTP 回传、中央调度控制和用户监控与交互等模块，并集成了数据自动处理模块。

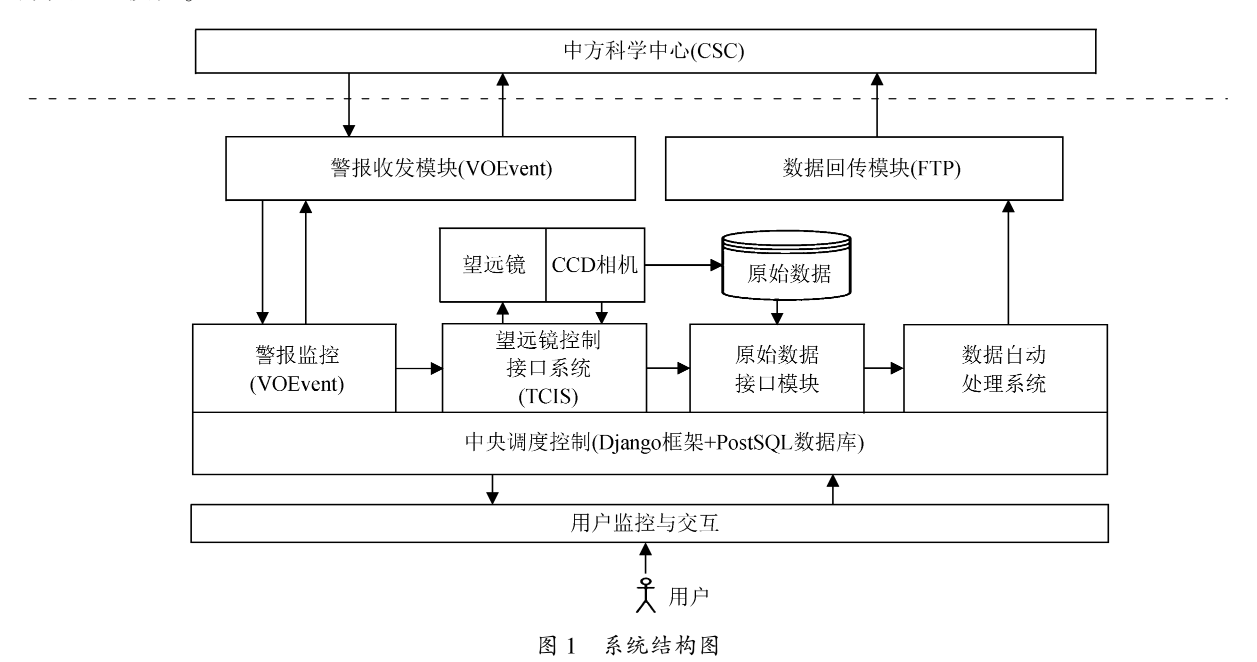


图 1　系统结构图

Fig. 1　System schematic

系统的开发基于 Python 语言，这是因为 Python 语言具有丰富的第三方天文软件开发工具包和友好的跨平台部署能力，以及面向对象的特性为系统开发和迁移带来极大的方便。基于 PostgreSQL 数据库的后台数据服务为系统调度的参数传递和数据产品的管理提供技术支撑。基于 Django 框架的网页服务为系统用户提供友好的人工交互接口与数据库交互接口。

Django 是一个开源的网络应用框架，其核心包含 URL 分发器、处理页面请求的视图系统、网页模板系统和一个数据模型与数据库之间的映射器。系统的用户监控与交互模块采用 Django 框架设计，用户通过浏览网页即可实时监视和控制系统。同时，Django 数据模型与数据库之间的映射器实现了数据库中表的对象化，为其他模块访问数据库提供了面向对象方式的读写操作接口。

系统研究与开发以中法天文卫星项目地基后随望远镜需求为基础，同时兼具处理不同望远镜数据的能力，即良好的可移植性。系统设计采用高内聚低耦合的软件工程设计思想，系统按功能划分为五大模块：中央调度控制，警报和数据实时传输与监控，数据自动处理与管理，硬件控制接口以及用户监控与交互。

2 实时消息与数据传输

针对中方科学中心(Chinese Science Center, CSC)和瞬变源快速自动响应观测系统之间的消息与数据通路需求，设计了两条传输路径分别传送不同类型的数据：(1)基于虚拟天文台事件(Virtual Observatory Event, VOEvent)的即时数据通讯，中方科学中心与瞬变源快速自动响应观测系统之间部分消息需要即时传递，采用这种通讯可以在秒级传递消息；(2)基于文件传输协议(File Transfer Protocol, FTP)的大数据传递通道，相比第1条通道，这条通道是用于瞬变源快速自动响应观测系统向中方科学中心传递数据量大且不需要即时传输的观测科学数据，如普适文件传输系统(Flexible Image Transport System, FITS)文件等。

VOEvent 是天文瞬变源观测领域内标准数据包的一种标准格式，包含多种信息如坐标、时间等，所有的信息封装在可扩展标记语言(Extensible Markup Language, XML)文件中，VOEvent 还可用于望远镜指向控制、传递观测计划、发送观测结果等等[7]。根据 VOEvent 的要求规范，定义了地基后随望远镜 VOEvent 参数。为保证消息和数据传递的可靠性，在设计上采用消息应答机制，所有信息在收到后均发送应答给发送方。传输的流程如图2，中方科学中心在收到伽玛暴触发信息后，发送警报到瞬变源快速自动响应观测系统，系统收到警报后对警报进行应答，同时开始处理警报内容并控制望远镜观测，在曝光完成后，系统自动处理观测原始数据，并发送一次 VOEvent(图中③)，包括是否是伽玛暴事件、星等、精确坐标等信息，如果发现伽玛暴，VOEvent 中还包含 Fchart(Finding Chart)信息；在结果 VOEvent(图中⑤)发送完成后，进行大文件的 FTP 上传，上传前发送 VOEvent 告知科学中心 FTP 上传的内容信息。此过程不断重复，直到观测结束。VOEvent 通路基于 Comet[8]项目的警报传输协议(VOEvent Transport Protocol, VTP)①，可以保证数据的完整性。

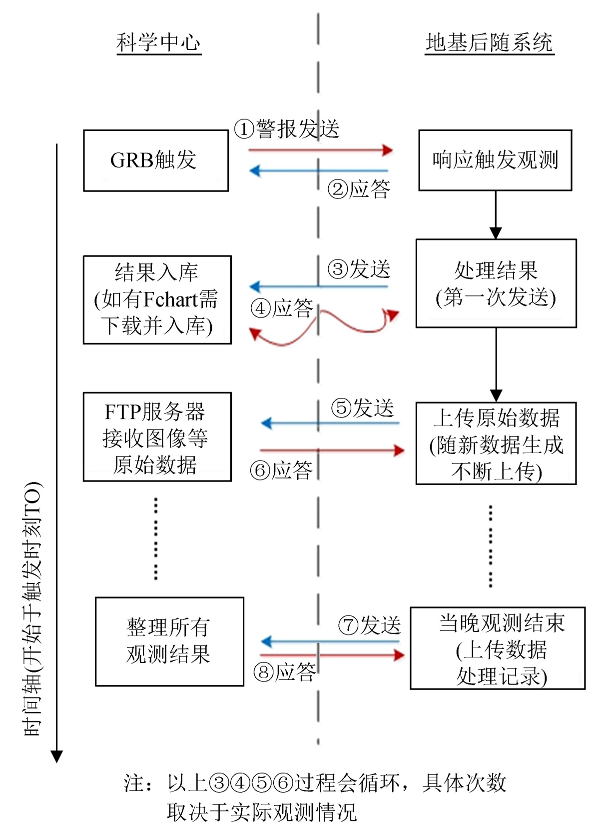


图2 VOEvent 传递图

Fig. 2 VOEvent transfer diagram

FTP 是一个客户机和主机之间文件双向传输协议②，FTP 数据服务器位于中方科学中心。数据传输使用密码验证、数据加密以及 md5 验证码验证的传输方式，保证了数据传递的可靠性与安全性。在利用 FTP 进行大数据图像文件传输的同时，会发送 VOEvent 消息告知中方科学中心所发送文件的属性，便于对所传文件的内容进行查收与确认。

3 中央调度控制

系统模块之间耦合强弱取决于模块间接口的复杂程度、进入或访问一个模块的点以及通过接口的数据。为降低系统的复杂程度，提高设计质量，采用数据库的方式解决模块间参数传递问题，模块间

① http://www.ivoa.net/documents/Notes/VOEventTransport/20090805/NOTE-VOEventTransport-1.1-20090805.html

② https://en.wikipedia.org/wiki/File_Transfer_Protocol

只需传递简单的数值即可完成通讯，更多的数据信息各模块可从数据库单独调取，这样极大地减少了模块之间的耦合程度。模块间参数采用数据库传递的设计，主要考虑相关系统参数存入数据库方便监控模块对系统状态的实时跟踪和分析，这有助于对系统的自动化管理和监控，增加了系统的透明性，提高了系统的稳定性，同时实现数据产品的管理。模块的执行不存在高并发情况，同时从设计上避免了两个模块对一个数据均有写入权限情况的可能性，从而避免数据库在参数传递中的弊端。系统在试运行过程中不断成熟，未来进一步研究消息中间件在参数传递中的应用。

整个系统的运行以中央调控为中心。如图 3，警报监控模块在监控到警报后，调用 Django 框架下的函数在数据库中新建记录，以 Trigger_ID 为主键写入警报信息，启动望远镜控制、原始数据接口、数据处理三大模块；望远镜控制模块在收到触发警报后，根据 Trigger_ID 从数据库中读取警报信息及观测策略信息，以便控制望远镜观测。原始数据接口模块根据数据库中相关表信息读取配置文件并启动监控，并把更改信息写入数据库。数据处理系统根据 Trigger_ID 从数据库读取定标文件、处理路径等信息，处理完成后，在数据库中新建相关记录并写入信息。

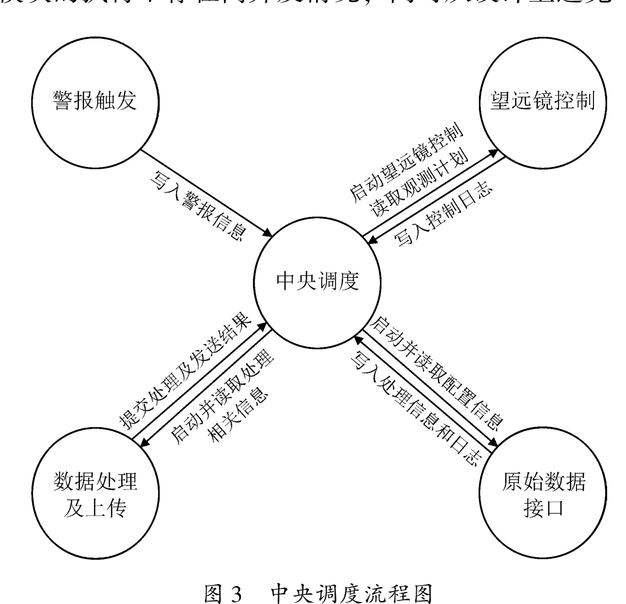


图 3　中央调度流程图

Fig. 3　Center scheduling flow chart

数据的自动化处理需要对数据进行管理。数据库数据类型大致可以分为以下几类：(1)系统状态信息，这类信息主要以键值的形式存于数据库，包括观测计划、系统运行状态、望远镜配置信息等；(2)定标文件及科学产品，这一类数据以 FITS 图等较大数据为代表，系统采用的方案是数据本身存储于硬盘，而对数据文件名、路径等信息写入数据库，程序在读取文件时，首先访问数据库，读出路径及文件名信息，再访问硬盘读取数据，对于一些关键数值如星等值等，则以数值的形式，记录在数据库中。系统通过 Django 提供的接口及 psycopg2[③] 对 PostgreSQL 数据库进行操作，除更改观测计划时间表需要通过网络接口手动修改外，其他信息均由系统自动管理，这种方式大大提高了系统的自动化程度。

根据系统的主要功能需求，数据库一共建立了 10 张表。数据库实体-联系图(Entity Relationship Diagram, E-R)如图 4(图中只标注了主键和外键)，实体-联系图中各表具体说明见表 1。

表 1　数据库表单说明

Table 1　Database forms introduce

表名	说明
obs_inf	观测信息表，记录观测基本信息
gft_status	系统状态表，记录当前系统状态
obplan	观测计划表，存储望远镜自动观测计划
raw_cal_images	原始定标文件表，记录原始定标文件信息
cal_im2_aufiles	定标文件附表，原始定标文件和定标文件之间关系表
cal_images2	定标文件表，记录定标文件信息
raw_sci_images	原始科学图像表，记录望远镜观测得到的原始科学图像相关信息
sci_images2	科学图像表，记录系统处理后图像文件信息
sci_im2_aufiles	科学图像处理附表，科学图像和定标文件关系对应表
sci_products	科学产品表，记录系统生成的科学产品信息

③　http://initd.org/psycopg/

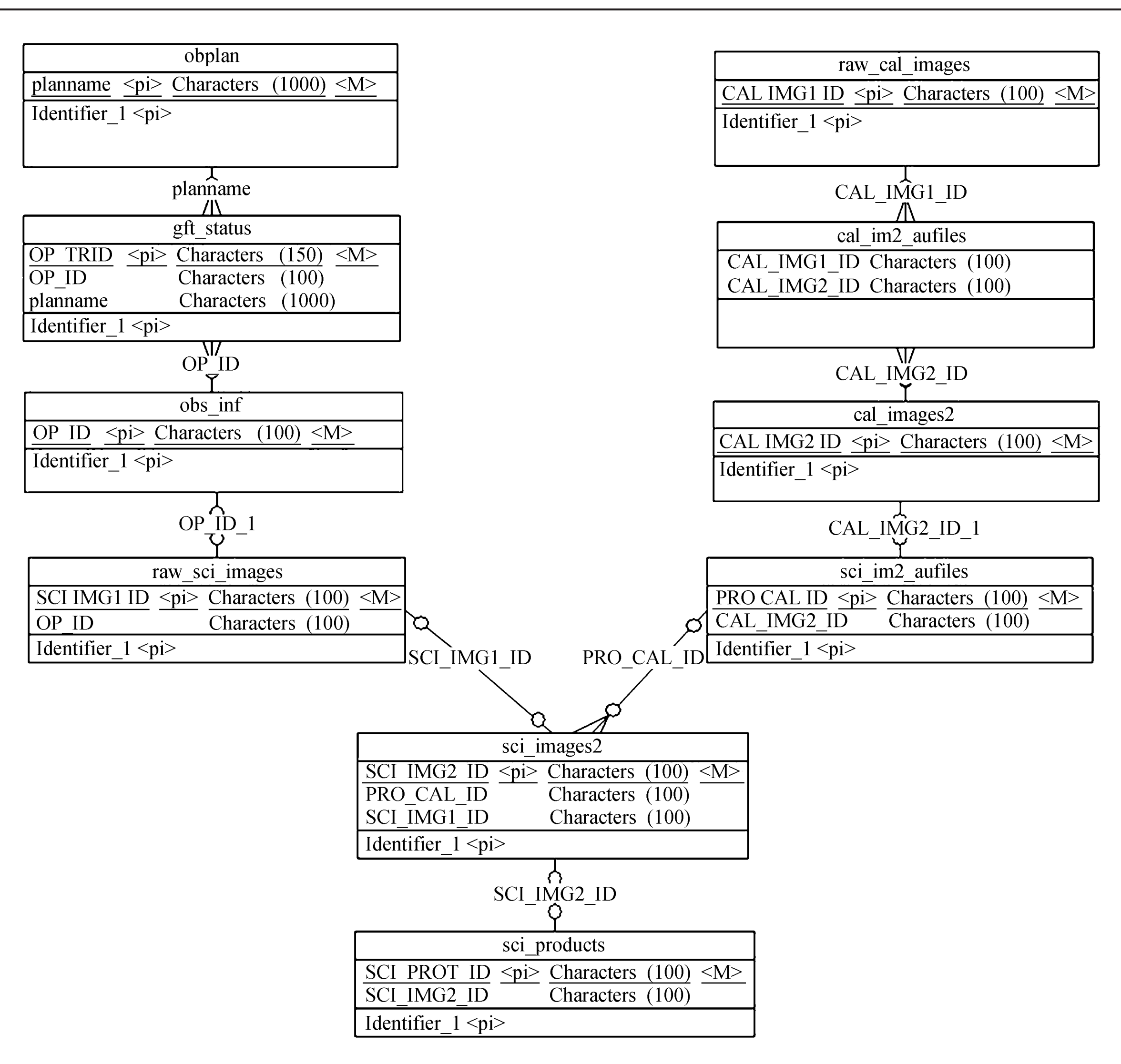


图 4 数据库实体-联系图

Fig. 4 Database ER Diagram

4 状态的监控与交互

为了工作人员能远程监测系统的运行状态以及人工方式对观测策略等进行修改，系统建立了状态监控与交互模块，该模块加入采用 B/S 网络架构技术的交互平台。平台基于 Django 框架，Django 采用了模型-视图控制器的框架模式，同时集成了一个轻量级的网络服务器，方便开发人员快速开发和调试④。

平台具有日志实时显示和观测策略制定两大功能。日志显示界面可以实时刷新监控系统运行日志信息，页面通过 AJAX 技术，在不重新加载整个页面的情况下与服务器进行数据交换，实现日志信息实时刷新的功能。观测策略制定页面分为 3 个板块：(1)显示当前系统已激活的观测计划，包括计划名称及具体时间安排；(2)数据库中已存储的观测计划名字列表及具体时间安排显示，方便观测人员灵活调用并激活；(3)增删观测计划，可根据实际观测需要，对观测计划进行调整。观测计划是以二维表的形式存于数据库，而每个观测计划包含计划名称、每个计划曝光时间顺序、每个顺序具体曝光时间的三维表。为有效管理数据库，减少数据存储空间和读取次数，把曝光时间顺序和每个顺序曝光时间通过编码合成一维。页面在提交数据时，在前端进行编码后传递。读取数据时，把编码的数据读

④ https://www.djangoproject.com/

取到前端后解码再显示。这样观测计划从数据库存取到传递，只需要一次操作即可完成读取或者存储一个观测计划的完整信息。

平台的使用基于浏览器，这让平台具有跨平台能力，用户可在不同操作系统上访问平台。用户在访问页面时，浏览器发出页面显示请求，Django 对请求进行处理，从数据库中读取和返回当前日志信息，并在页面显示。Django 处理请求运行简图如图 5。

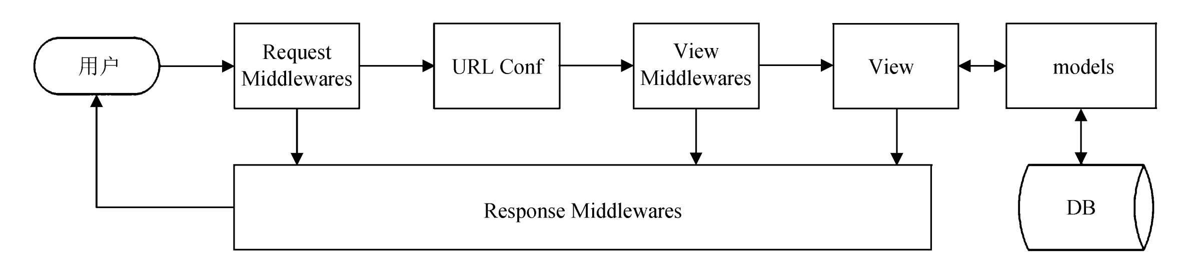


图 5　Django 流程图

Fig. 5　Django flowchart

交互平台目前已完成观测策略的制定和浏览页面，以及后台程序运行日志文件的实时监控页面的开发。用户访问平台时，通过在浏览器输入指定地址，Django 根据地址匹配调用相应视图函数，视图函数通过数据模型与数据库之间的映射器访问数据库。页面的静态显示通过加载模板实现，模板的编写基于 HTML、JavaScript 以及 CSS 样式，实际界面如图 6。

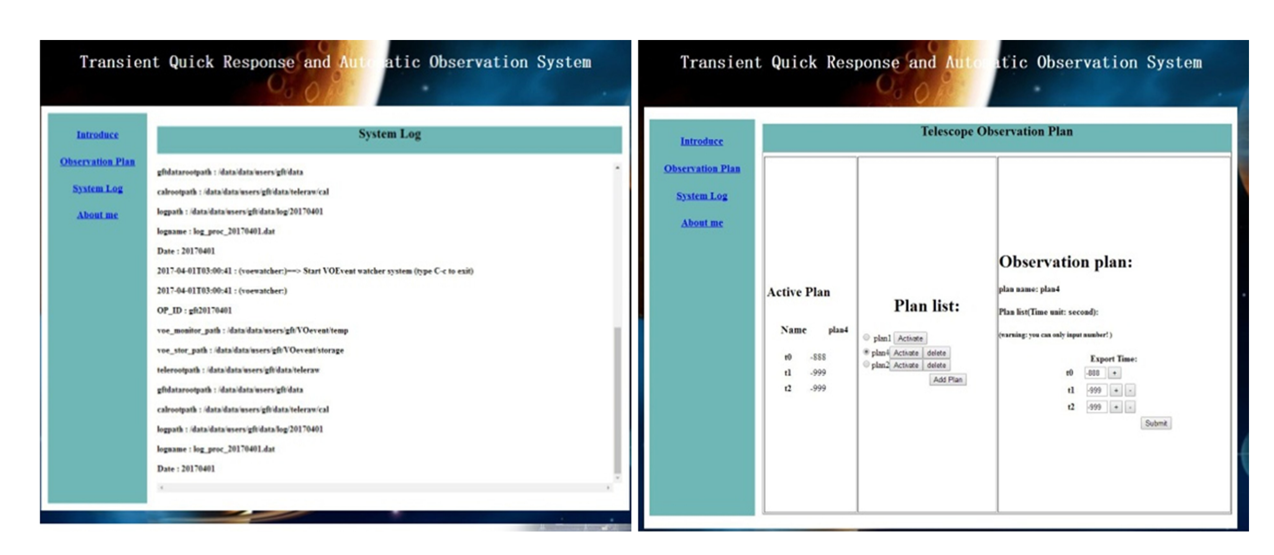


图 6　网络交互平台界面

Fig. 6　Interface of web interactive platform

5　系统测试

系统开展了功能和性能两方面的测试。功能测试包括警报发送、警报解析、望远镜自动控制观测、原始数据接口（接收原始数据）、数据处理及处理结果上传等功能。性能测试包含系统警报响应耗时、系统完成整个触发观测流程耗时以及数据处理结果正确性测试。

测试框架流程如图 7。根据实际应用场景，位于北京的科学中心仿真生成测试警报 VOEvent，通过警报传输协议发送到警报调度服务器，警报调度服务器发送警报至位于兴隆观测基地的瞬变源快速响应观测系统，观测系统解析后控制望远镜自动观测，完成数据采集后将采集的数据进行自动处理，处理完成后结果上传北京的科学中心。数据的传输主要用到两个链路：FTP 传输链路和 VOEvent 消息实时传输链路。

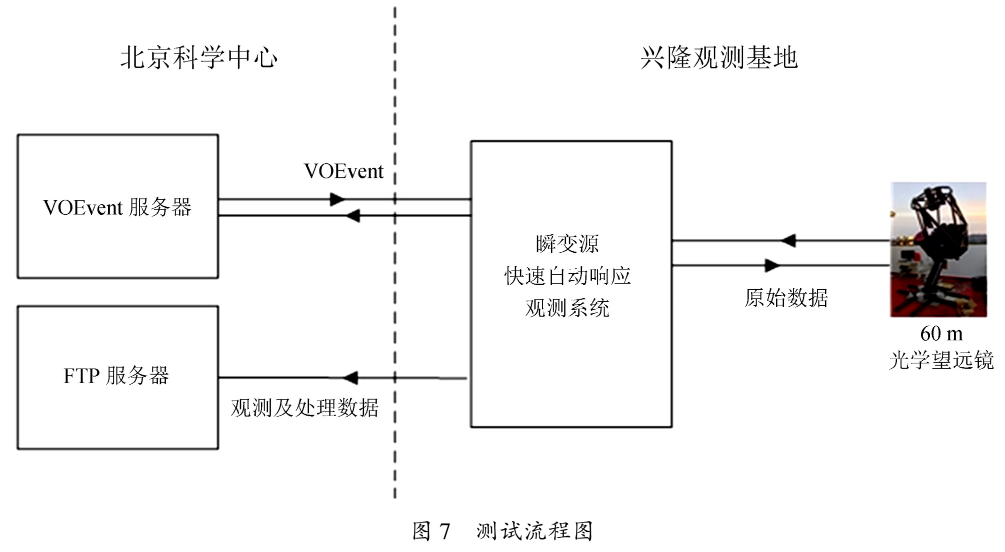


图 7 测试流程图

Fig. 7 Test flowchart

测试包括科学中心发送警报信息，系统触发并运行完成，分析系统日志以及返回给科学中心的结果数据。通过两个观测夜的多次测试，结果表明，系统无漏报或者错报现象，各个预期测试功能完全正常运行，FTP 服务器及消息 VOEvent 接收传递的数据完整无误。

通过测试表明：从警报发送到望远镜观测控制指令执行，系统所需要的响应时间平均约为 2 s。具体见图 8，即瞬变源快速自动响应观测系统在收到警报后，系统解析触发警报并控制望远镜观测耗时的统计图，图中数据来源于瞬变源快速自动响应观测系统多次实际测试观测生成的日志文件。

系统完成一个流程所需时间测试结果如图 9，即从多次观测中，统计瞬变源快速自动响应观测系统从收到警报到数据上传完成所需的时间。这个时间除了包括系统触发警报、数据处理等运行时间，还包括望远镜的曝光时间。从图中可以看出，在望远镜 30 s 曝光时间设定下，系统完成一个流程最大耗时不超过 90 s，如考虑实际应用指标 100 s 的曝光时间，则系统完成时间最大不超过 150 s。图中数据来自于瞬变源快速自动响应观测系统的日志文件。

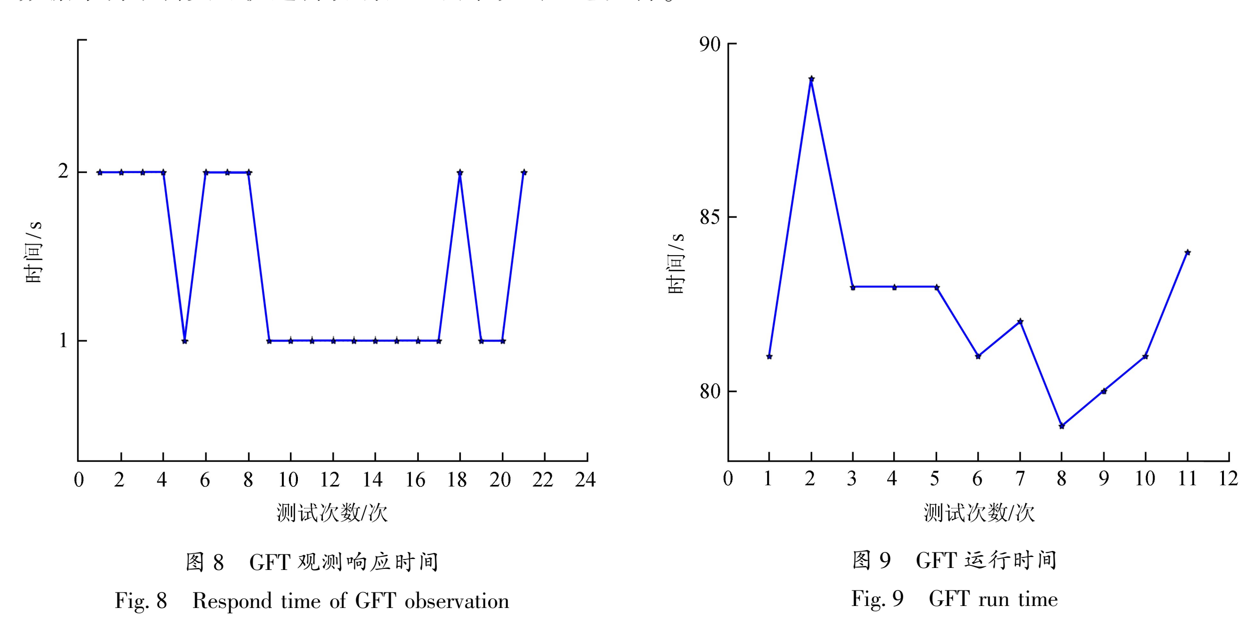


图 8 GFT 观测响应时间

Fig. 8 Respond time of GFT observation

图 9 GFT 运行时间

Fig. 9 GFT run time

为了检验观测执行的正确性，将上传的观测结果与 SAO-DSS 服务器上其它定标望远镜的观测结果进行对比。如图 10，左图是系统的自动观测结果，右图是 SAO-DSS 的结果。对比表明自动观测系统正确地执行了观测指令，观测图像具有科学性和准确性。

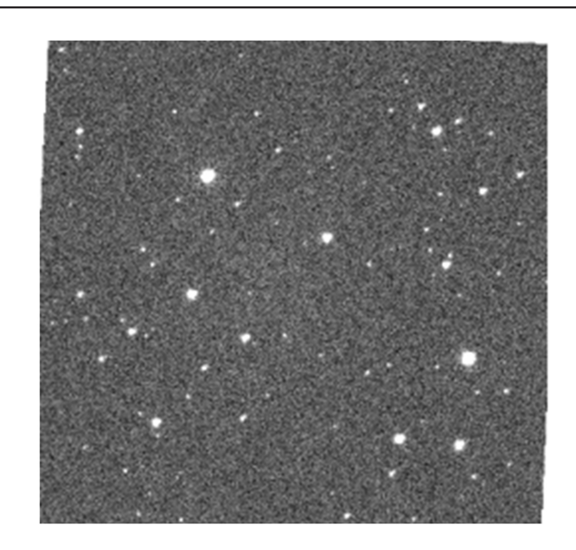
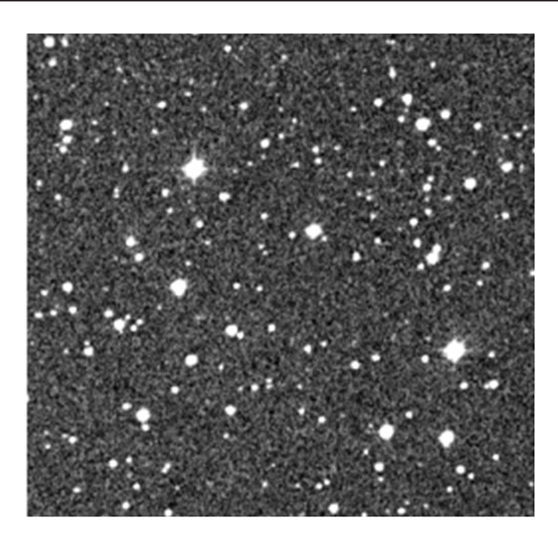



图 10 GFT 处理结果图及对应天区 SAO-DSS 巡天图像

Fig. 10 Exposure of C-GFT after triggering and Retrieved image from SAO-DSS server

6 结论与展望

瞬变源快速自动响应观测系统是根据 SVOM 项目地基后随望远镜对伽玛暴等天文瞬变源的观测需求开发的。系统具有如下功能：自动响应科学中心的观测警报；自动获取观测计划，根据观测计划的要求控制望远镜指向，并采集观测数据；自动启动瞬变源搜索处理程序；将处理结果和观测数据实时上传给科学中心。同时，系统提供人工监控与交互接口功能，以网页服务的方式提供用户进行人工监控系统运行状态，以及调整观测计划等。

系统主要基于面向对象的 Python 语言开发。Django 框架为系统提供了网页、数据处理等应用层与数据库间面向对象的友好的数据交互接口。以数据库为后台支撑，系统既实现自动处理模块间的调控参数传递，又实现了对数据及其结果的管理。以网页服务为基础的运行监控及交互方式极大提升了系统远程运行和人机交互的友好性。基于 Python 丰富的天文数据处理软件及 VOEvent 消息传输软件为系统的快速稳定运行提供保障。

根据地基后随望远镜运行的实际场景进行了瞬变源警报响应时间、数据观测结果上报时间以及数据响应的真实性验证测试。测试结果表明：系统对警报的响应耗时平均小于 2 s；从开始观测到观测结果上报耗时小于 160 s（以典型曝光时间 100 s 计）；观测结果图像与 SAO-DSS 的图像对比表明，系统进行了正确的指向和数据处理。

系统具有无人值守、实时监控、快速自动处理、数据实时传输等基本功能和特点。系统的鲁棒性设计有以下两方面：(1)在模块内部，发生现有程序设计之外的状况做抛弃处理，同时将模块状态写入数据库及日志，以便监控程序查看和调用；(2)各模块提供应急启动接口，可以在数据库连接出问题时通过直接传递参数方式跳过数据库启动模块。系统无特殊的依赖包，安装简单。通过望远镜控制接口部分的简单更改，即可移植到其他已运行的望远镜，并且系统输入输出清晰，内部运行独立于设备，因此具有较强的通用性。对于其它类似的瞬变源自动响应后随观测也适用。

致谢：感谢周丽晓老师对系统测试部分的建议与有益讨论。感谢 SVOM 项目组工程与天文观测人员在系统搭建与测试过程中给予的帮助。

参考文献：

[1] Paul J, Wei J, Basa S, et al. The Chinese-French SVOM mission for gamma-ray burst studies [J]. Comptes Rendus Physique, 2011, 12(3): 298-308.

[2] Castro-Tirado A J. Robotic astronomy and the BOOTES network of robotic telescopes [J]. Acta Polytechnica, 2011, 51(1): 16.

[3] Tyurina N, Lipunov V, Kornilov V, et al. MASTER prompt and follow-up GRB observations [J]. Advances in Astronomy, 2010, 2010: 1-6.

[4] Akerlof C W, Amrose S, Balsano R, et al. ROTSE all-sky surveys for variable stars. I. test fields [J]. The Astronomical Journal, 2000, 119: 1901-1913.

[5] Guidorzi C, Monfardini A, Gomboc A, et al. The automatic real-time Gamma-Ray Burst pipeline of the 2m Liverpool Telescope [J]. The Publications of the Astronomical Society of the Pacific, 2006, 118(840): 288-296.

[6] Vestrand W T, Borozdin K, Casperson D J, et al. RAPTOR: closed-loop monitoring of the night sky and the earliest optical detection of GRB 021211 [J]. Astronomische Nachrichten, 2004, 325(6-8): 549-552.

[7] Zhang Mo, Huang Maohai, Wu Chao. Prototype VOEvent network systems based on VTP and XMPP for the SVOM Chinese science center [C]// 14th International Conference on Space Operations. 2016.

[8] Swinbank J. Comet: a VOEvent broker [J]. Astronomy and Computing, 2014(7): 12-26.

Research and Implementation of Transient Quick Response and Automatic Observation System

Fu Xiachuan[1,2,3], Wu Chao[1,2], Huang Maohai[1,2,3], Zhang Mo[1,2], Lu Xiaomeng[1,2]

(1. National Astronomical Observatories, Chinese Academy of Sciences, Beijing 100012, China, Email: cwu@bao.ac.cn;
2. Key Laboratory of Space Astronomy and Technology, National Astronomical Observatories, Chinese Academy of Sciences, Beijing 100012, China; 3. University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Follow-up observation of astronomical transient sources is important for the identification of the transient source. The random nature of the transient sources and the speed of change of brightness require the follow-up observing system to react quickly and automatically. In this paper we describe our study and implementation of the Transient Quick Response and Automatic Observation System based on Python. At the core is the functional realization of database-interface and module-scheduling based on Django architecture. Main functions of this system include alert message transmission using VOEvent, a web-based status monitoring and interface automatic data processing and reporting, and telescope control interfaces. Tested with data transfer between the SVOM Chinese Science Center located in Beijing and Xinglong Observation Station, and actual triggered observation, the system has met the basic automatic transient follow-up observation requirements of the SVOM project. At the same time, the system can be used for other similar automatic response observation thanks to its portability.

Key words: Astronomical data processing; Transient source; Automatic observation